Ｒ８Ｃ／Ｍ１２Ａ　　防犯ブザー　取扱説明書

第１版２０１１．１２．５　　**第１版**

【　製品概要　】

防犯ブザーはルネサスエレクトロニクス社の２０ピンＤＩＰマイコン、Ｒ８Ｃ/Ｍ１２Ａを使用し、振動を検地し、アラームを鳴らす防犯グッズです。

電源投入すると、１０秒間は無動作期間があります。この間にアラームから離れてください。(緑ＬＥＤ点滅)、以降、赤ＬＥＤが点滅し、振動検出を行います。振動を検出すると、２秒後に再び振動を監視し、継続している場合、１０秒後、ブザーを鳴らします。継続していない場合、振動検出に戻ります。

特徴は以下の通りです。

１．ご家庭の玄関、窓際や車の中などで防犯ブザーとして使えます。
　キットではなく完成品です。単三電池２本ご用意いただければ、すぐに動作します。

２．単３電池２本でも、連続８３日間監視　※１　しますが、別途ＵＳＢから電源を供給することができます(ＵＳＢミニＢコネクタ搭載)。ＡＣ１００ＶからＵＳＢ（５Ｖ）変換アダプタや、車の場合、市販のシガーソケット（１２、２４Ｖ）→ＵＳＢ変換（５Ｖ）アダプタが使用できます。
※振動が継続する場所での正しい動作はできません。異常時以外、振動の無い場所でお使い願います。
※１　本体消費電流 1mＡ、単３アルカリ電池　２０００ｍＡ／ｈでの計算値。

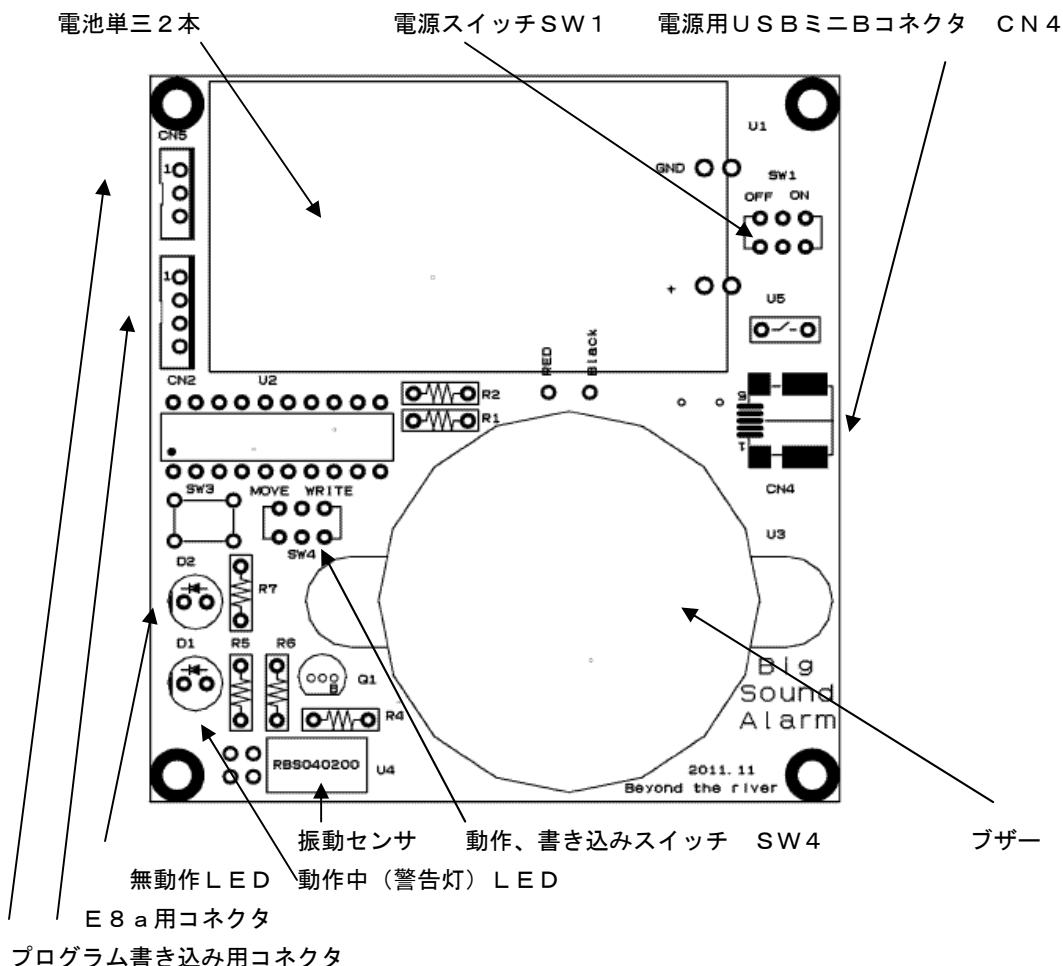
以下はマイコン学習に最適な特徴です。
３．動作プログラムが解説されているので、プログラム作成の方法が理解ができます。(Ｃ言語で記述されています)
４．別売の書き込み用Ｖケーブルを用意すれば、他は無償の開発環境でプログラムを好きなようにカスタマイズできます。
５．Ｅ８ａを使用した、ブレークポイントや変数値を参照できる高度なデバックにも対応できます。

本マニュアルは　振動アラームの使い方、プログラム動作解説、プログラム開発を行うために必要なソフトウエアインストゥール手順、について解説されています。

１．使い方

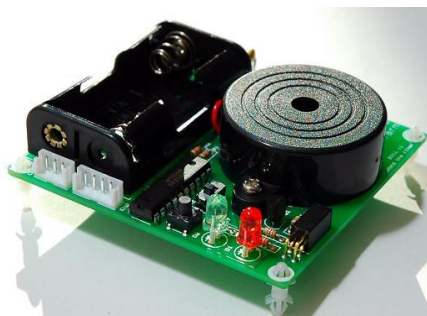
１－１．　各部の機能と名称
１－２．　使い方

１．使い方

１－１．　各部の機能と名称

本機の消費電流は　待機時 0.32mA、LED 点灯時 1.6mA、ブザーON 時　3.3mA 程度です。(いずれも 3.3V 電池での実測値)。ですので、待機時消費電流は 0.32+（(1.6-0.32）/3.1）≒　0.73mA となります。おおよそ１mA です。

※3.1　は後にプログラムの解説で出てきますが、警告灯が ON する時間の電流消費＝1.6mA、時間＝0.1 秒、点灯しない時間の電流消費＝0.32mA、3 秒ですので、LED が点灯したときの突出した電流＝1.6-0.32 を連続した電流としてならした計算を（1.6-0.32）/3.1 として、待機電流に加算しています。

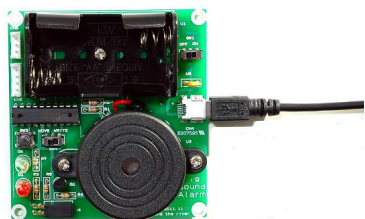
１－２．　使い方

a. 添付の足を写真のように４隅に取り付けてください。穴に足を入れると、抜けなくなります。基板の裏は回路の配線、半田付け部分の凹凸があり、この部分に水や金属など、導電物が接触すると正しく動作できませんので、写真のように下から浮かして使用してください。

b. 電池を使用する場合は単３電池２本を正しい方向に入れてください。ＵＳＢ　ミニコネクタから供給する場合、必ず電池ははずしてください。電池を入れたままＵＳＢ側から電源を供給すると、過電流が流れ、発熱したり、火災になる可能性があります。

ｃ．ＭＯＶＥ/ＷＲＩＴＥスイッチをＭＯＶＥ側

ｄ．電源スイッチＳＷ１をＯＮ

緑の無動作ＬＥＤが１０秒間点滅します。この間にアラームから離れてください。以降、赤ＬＥＤが点滅し、振動検出を行います。振動を検出すると、２秒後に再び振動を監視し、まだ振動している場合、事故発生として１０秒後、ブザーを鳴らします。継続していない場合、振動検出に戻ります。

セット後、ＯＦＦするときは触ると振動検出しますが、１０秒間はブザーは鳴りませんので、その間に電源スイッチをＯＦＦしてください。

ＵＳＢから電源を取る場合　市販の USB ミニコネクタが使用できます。　電池は絶対に入れないで下さい。

２．プログラム動作解説

```
/************************************************************************/
/*                                                                      */
/*  FILE        :R8C_alarm.c                                            */
/*  DATE        :Thu, Dec 01, 2011                                      */
/*  DESCRIPTION :main program file.                                     */
/*  CPU GROUP   :M12A                                                   */
/*                                                                      */
/*  This file is generated by Renesas Project Generator (Ver.4.19).     */
/*  NOTE:THIS IS A TYPICAL EXAMPLE.                                     */
/************************************************************************/

①#include "sfr_r8m12a.h"


//割り込み処理
②#pragma interrupt _timer_rc(vect=7)

③volatile unsigned char cdata,sw_level,sindou_flg,sindou_cnt;
volatile unsigned short int_timer,loop;

④#define D1_ON            p1_7 = 0
#define D1_OFF           p1_7 = 1

#define D2_ON            p1_2 = 0
#define D2_OFF           p1_2 = 1

#define BUZZER_ON  p1_3 = 1
#define BUZZER_OFF       p1_3 = 0


⑤void _timer_rc(void)//約 10msec
{

//       p1_0 = 1;          //マーカー

        imfa_trcsr = 0;                         //IMFA flag clear
         if(int_timer != 0){int_timer--;}

//    p1_0 = 0;             //マーカー
}
```

【　上記までの解説１　】

①#include "sfr_r8m12a.h"
　これはポートの絶対アドレス等を記入したヘッダファイルで、必ず、この１行を加える必要があります。

//割り込み処理
②#pragma interrupt _timer_rc(vect=7)

本プログラムではタイマーで定周期割り込みをかけています。その割り込み発生時に実行されるプログラム名です。

③volatile unsigned char cdata､､；
これは本プログラムで使用している変数の指定です。Volatile が無いと、コンパイラのほうで勝手にコンパイルしない可能性があります。

```
④#define D1_ON            p1_7 = 0
#define D1_OFF            p1_7 = 1
```

LED の D1 を ON にするためには p1_7 ポートを０にする必要があります。(別紙回路図参照) このように定義するとそれをプログラムでは D1_ON と書けばよくなりますので、間違いが少なくなる上に、覚えるのが簡単です。

```
⑤void _timer_rc(void)//約 10msec
{

//        p1_0 = 1;          //マーカー

        imfa_trcsr = 0;                     //IMFA flag clear
         if(int_timer != 0){int_timer--;}

//    p1_0 = 0;              //マーカー
}
```
②で定義した割り込み関数が⑤です。約 10msec に１回、メインプログラムが実行中にも関わらず、割り込んで実行されます。中でおこなっていることは割り込みのときに立ったフラグのクリアと、メインルーチンで使うタイマーint_timer の減算です。

以下省略

３－１．　開発環境の用意

ａ：無償版ＨＥＷ、Ｒ８Ｃ用Ｃコンパイラのダウンロード

　プログラムの開発は例えばルネサスエレクトロニクス社の統合開発環境ＨＥＷでＣ言語を用い動作させることができます。ＣＤ添付のサンプルプログラムはこの環境下で作成されています。無償版をダウンロードして使用します。

　ネット検索で→ルネサス　マイコン　Ｒ８Ｃ　と入力し、関連リンク→無償評価版ソフトウエアダウンロードをクリック。

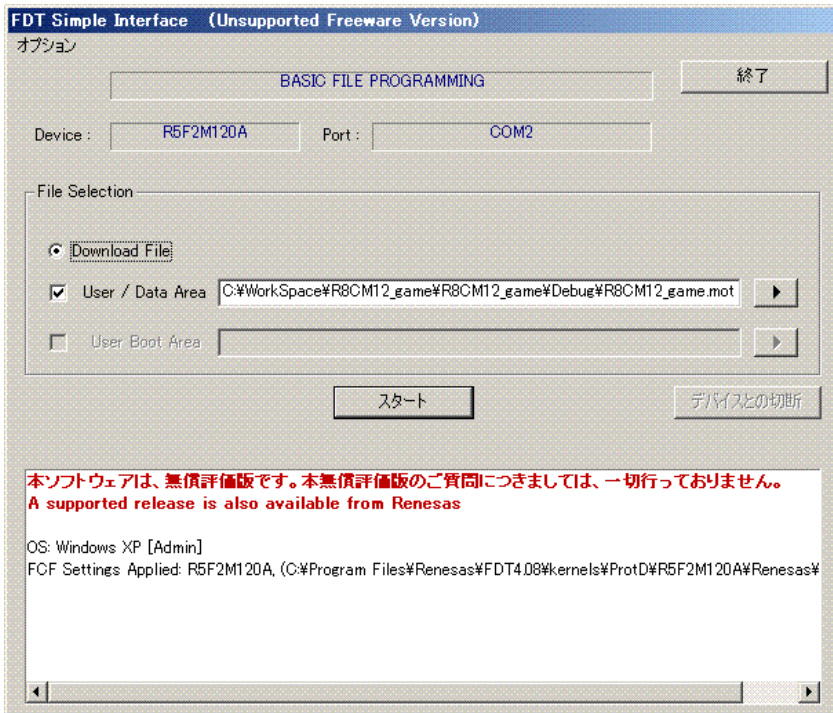
→Ｍ１６Ｃシリーズ、Ｒ８Ｃファミリ用　Ｃ／Ｃ＋＋コンパイラパッケージをダウンロードします。

	M16Cシリーズ, R8Cファミリ用 C/C++コンパイラパッケージ (M3T-NC30WA) 製品ページ 評価版ダウンロード 製品版をご購入の際は、R8C, M16Cファミリ用C/C++コンパイラパッケージをご注文ください。詳細は 製品ページを参照ください。	V.6.00 Release 00 • 試用期限内は製品版と同じ。 • 試用期限を過ぎるとリンクサイズが64Kバイト以内に制限されます。 • **High-Performance Embedded Workshopおよびシミュレータデバッガを同梱。**	60日 初めて評価版ソフトウェアツールをインストールした後、最初にビルドを行った日から60日。 *一度でもビルドを行った場合は評価版ソフトウェアツールをインストールしなおしても試用期限の延長はできません。

　無償版は６０日経過後、リンクサイズが６４ＫＢと制限されますが、ＲＯＭ容量２ＫＢのＲ８Ｃ／Ｍ１２Ａにとっては事実上、有償版と違いがありません。
統合開発環境ＨＥＷとＣコンパイラがインストゥールされます。

ｂ：書き込みツールＦＤＴのダウンロード

Ｒ８Ｃ/Ｍ１２にプログラムを書く方法は
１．ルネサスのサイトからＦＤＴを無償ダウンロードして使用する。
２．Ｅ８ａを使う
の二つがあります。ここではＦＤＴで書く方法を説明します。別売の弊社Ｖケーブルが必要です。

ＦＤＴ　ダウンロード　で検索すると画面が表示されますので、ダウンロードしてください。
２種類のＦＤＴがありますが、ＣＯＭポートを使うＢＡＳＩＣを使用します。
画面を以下のように設定します。変更は「オプション」を操作することで可能です。
ＣＯＭポートの確認、変更は別紙「ＣＯＭ番号を調べる」を参照下さい。下例ではＣＯＭ２で設定しています。

詳細はダウンロード添付のマニュアルrjj10i0142_0806um.pdf「M16C Flash Starterユーザースマニュアル」をご参照下さい。

ｃ：プログラム、デバイスドライバのダウンロード、インストゥール

購入時、添付されるダウンロードパスワードで、弊社Ｒ８Ｃサイト（beriver.co.jp/R8C.html）よりプログラムとデバイスドライバのダウンロードを行います。（パスワードが必要です）

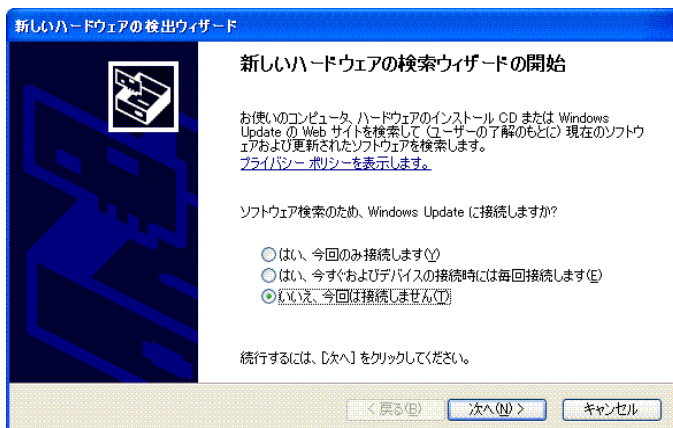
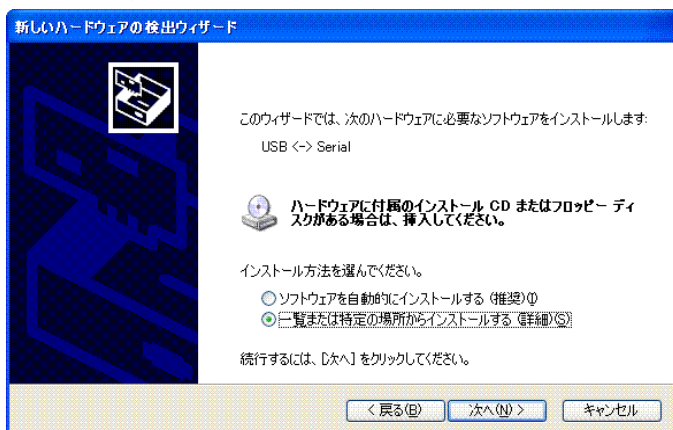
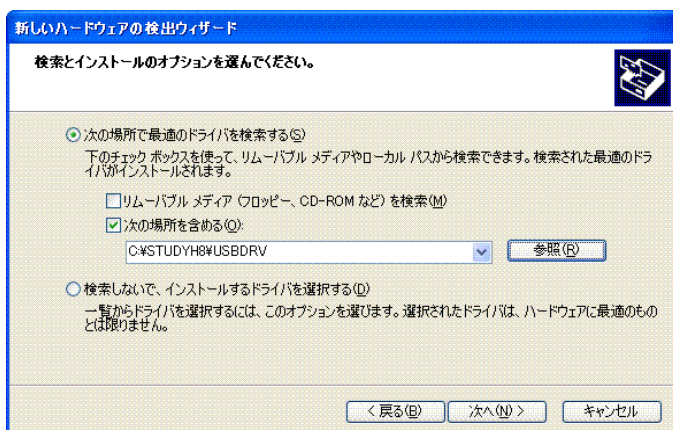
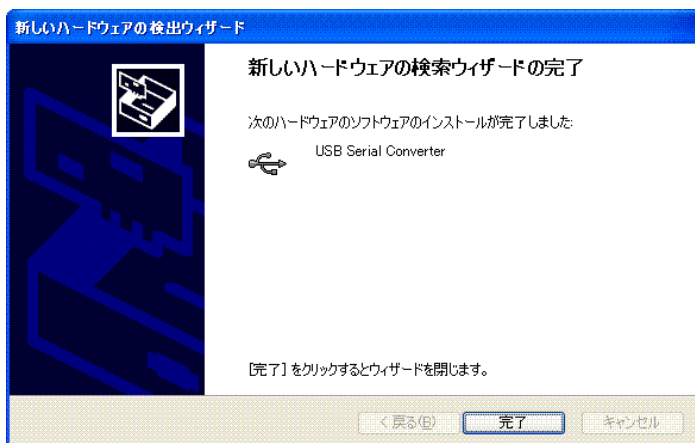
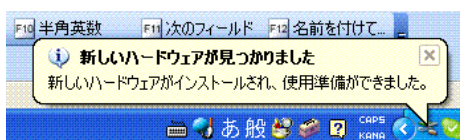
自己解凍ですので、ダウンロードされたファイルをクリックしてください。以下のホルダが展開されます。

取扱説明書	ファイル フォルダ	2011/12/06 16:21
回路図	ファイル フォルダ	2011/12/06 16:21
USBDRV	ファイル フォルダ	2011/12/06 16:21
R8Cマニュアル	ファイル フォルダ	2011/12/06 16:21
R8C_alarm	ファイル フォルダ	2011/12/06 16:21

プログラムＲ８ＣＭ１２＿ａｌａｒｍはC:¥WrokSpace¥にコピーしてください。WorkSpace は HEW をインストゥールすると自動形成されます。

次に、プログラムの書き込をＶケーブルで行う場合、デバイスドライバを以下の手順でインストゥールして下さい。

初めて、ＶケーブルをパソコンにＵＳＢミニケーブルで接続すると OS がデバイスドライバを要求してきます。※１

「新しいハードウエアが検出されました」と表示され、「新しいハードウエアの検出ウィザードの開始」が表示されます。デバイスドライバの設定を行います。（下記例は WindowsXP のウィザード例）
WindowsUpdate への接続は「いいえ、今回は接続しません」を選択し、「次へ（Ｎ）＞」をクリックしてください。

「一覧または特定の場所からインストールする（詳細）（Ｓ）」を選択し、「次へ（Ｎ）＞」をクリックしてください。

通常のインストゥールでは「参照（Ｒ）」をクリックし C:¥WrokSpace¥USBDRV を選択します。「次へ（Ｎ）＞」をクリックしてください。

インストールが正常に終了したら「新しいハードウエアの検索ウイザードの完了」が表示されますので、「完了」をクリックしてください。その後、再びウイザードが立ち上がりますが、同じように繰り返してください。(仮想 COM ドライバおよびダイレクトドライバ D2XX　インストゥールで２回行います)

「新しいハードウエアがインストールされ、使用準備ができました」と表示されたらＯＫです。

ここでは仮想ＣＯＭドライバを使用します。ＵＳＢにＣＯＭを仮想的に割り振る使い方です。Ｖケーブルが ＣＯＭの何番にインストゥールされたか、変更等は別紙「ＣＯＭ番号を調べる」を参照下さい。

これでＵＳＢの初期設定は終わりです。次回からはＵＳＢケーブルを挿入すれば仮想ＣＯＭとして認識され動作します。

以下省略

３－２　　ＨＥＷコンパイル、フラッシュＲＯＭ書き込み

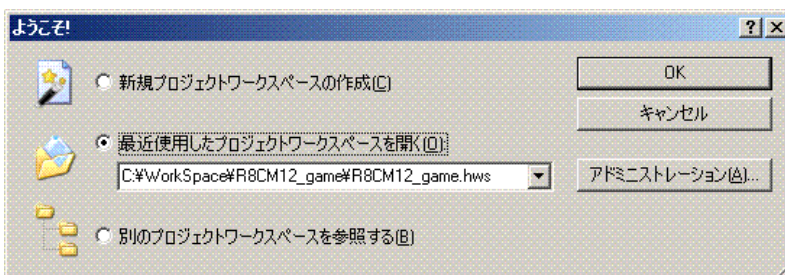
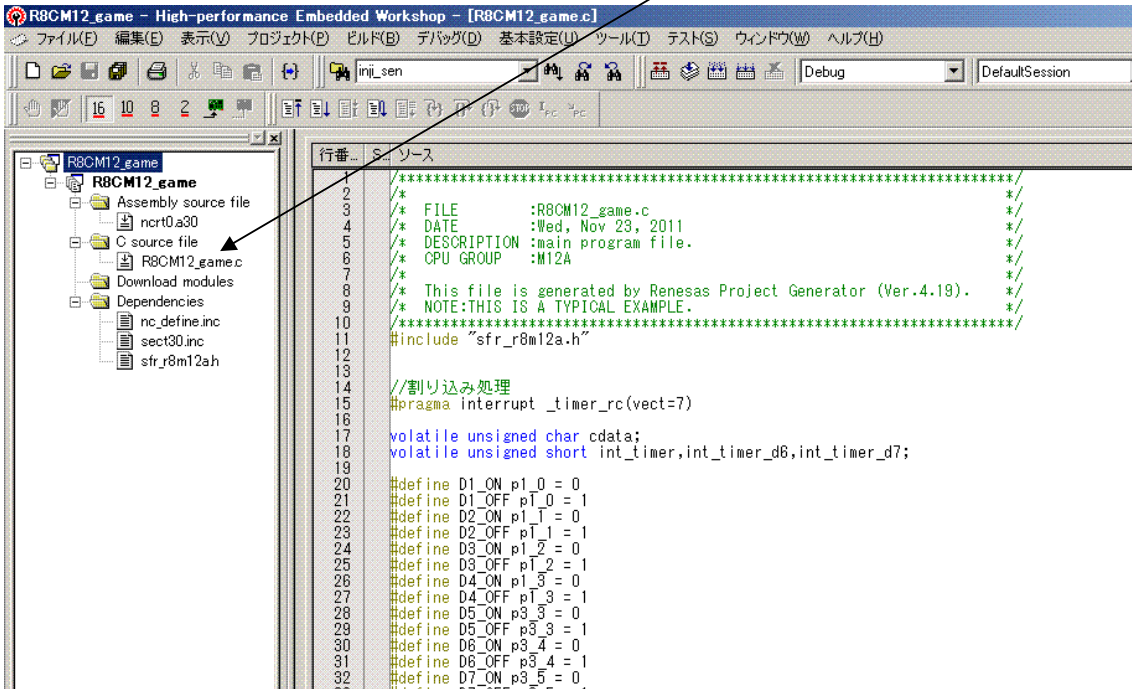
ａ：フラッシュＲＯＭライタ準備

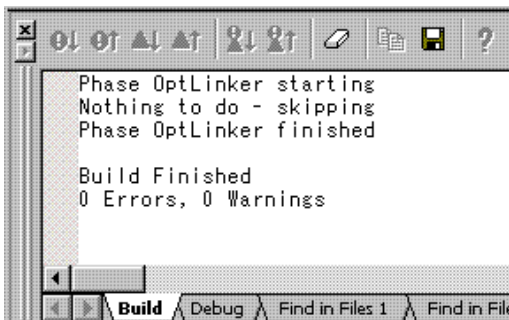
インストゥールしたＦＤＴ　ＢＡＳＩＣのショートカットを表画面に出します。

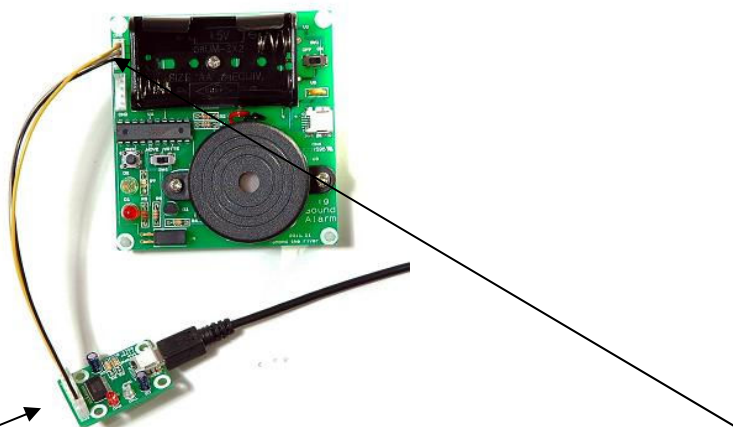
ｂ：ＨＥＷ起動、コンパイル、書き込み、動作

ＨＥＷを起動します。

初めてのときは、「別のプロジェクトを参照する」をクリックし、「ＯＫ」をクリックして C:¥WorkSpace¥にセーブした C:¥WorkSpace¥R8C_alarm¥R8C_alarm.hws を選択します。次回からは「最近使用した、、」を選択し、「ＯＫ」をクリックするだけで使用できます。

以下、R8CM12_game でサンプルを示しますが、R8C_alarm に置き換えてください。

ソースファイルが表示されます。表示されない場合、Ｒ８ＣＭ１２＿ｇａｍｅ.ｃをダブルクリックします。

プログラムのコンパイルは、以下のボタンで行います。ボタンはマウスを乗せると意味が表示されます。

コンパイルするとＲ８ＣＭ１２＿ｇａｍｅは出荷時にすでにコンパイラされていますので、０　Errors、０　Warnings と表示されます。新たにプログラムを製作してコンパイルしたときに Errors が０で無い場合、プログラムに文法上の問題がありますので、エディタでソースファイルを修正し、Errors　０にする必要があります。Errors が０で無い場合、書き込み用ファイル xxx．ｍｏｔが新たに作成されません。前のファイルは残りますので、勘違いして昔のファイルを書き込んでしまうことがあるので注意が必要です。

以下省略

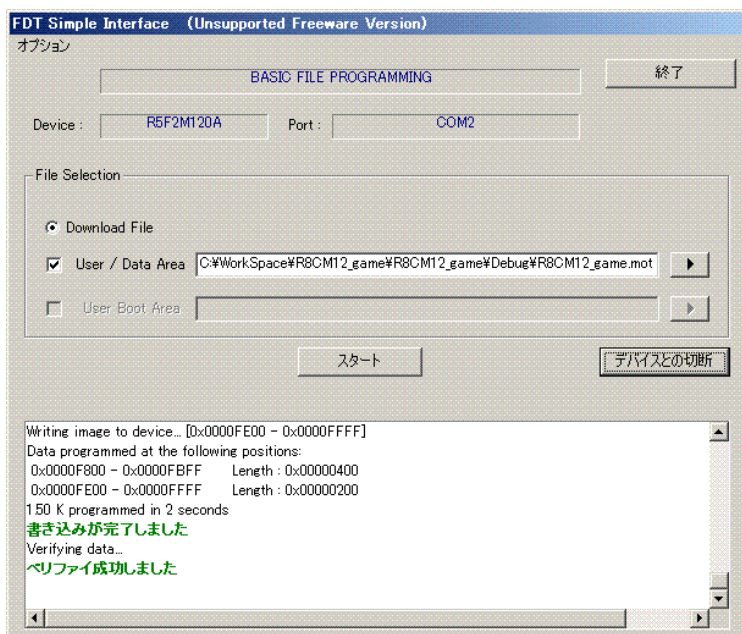
ｃ：ＦＤＴでプログラムを書き込む

ＶケーブルをパソコンからのＵＳＢケーブルに挿入し、Ｒ８Ｃアダプタをゲーム機に接続します。電源は電池を挿入してください。

１．電源スイッチＳＷ１はＯＮ

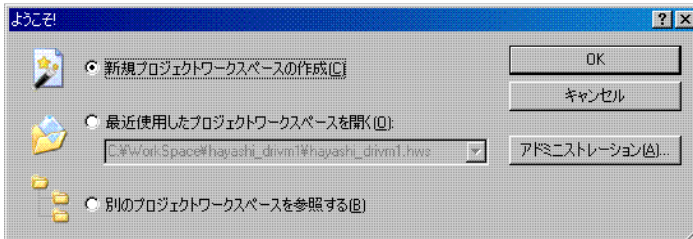
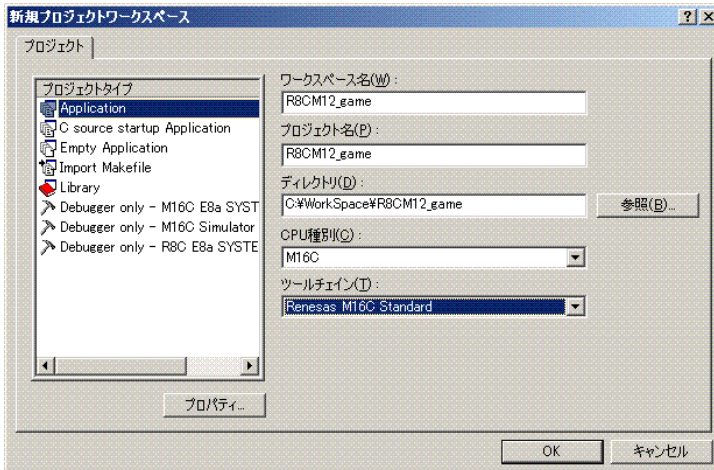
２．動作切り替えＳＷ３はＷＲＩＴＥ側

３．ＲＥＳＥＴスイッチＳＷ２を１回押す

これで準備完了です。

書き込むファイルは

C:¥WorkSpace¥R8C_alarm¥R8C2_alarm¥Debug¥R8C_alarm.mot です。

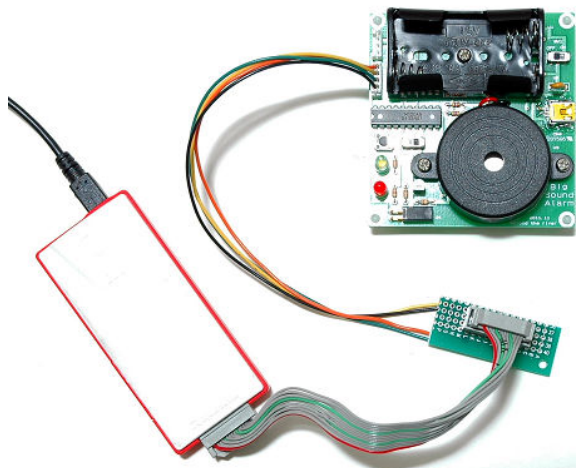
ｅ：新しいプログラムを作る

下記の例では WorkSpace に R8CM12_game というプログラムを作成する、という前提で作成しています。

CPU 種別、ツールチェインは M16 を選択します。

R8C　M12A を選択します。

以下省略

３－３．　Ｅ８ａでの動作、デバック

ａ．Ｅ８ａについて

Ｅ８ａ本体

Ｅ８ａとＲ８Ｃ＿振動アラームの接続例

　Ｅ８ａはＵＳＢ接続で使用する、ルネサスセミコンダクタ社のマイコン書き込み器、ＪＴＡＧデバッカです。書き込み器としてはＳＨマイコンからＴｉｎｙマイコンまで書けます。デバッカとしてはＨ８／Ｔｉｎｙシリーズ、Ｒ８シリーズなどでＣ、アセンブラソースデバックに対応します。ブレークポイント設定やメモリ、Ｉ／Ｏの読み込み、書き込みができます。

さらに、ＵＳＢバスパワーで最大３００ｍＡまでマイコンに電源（３．３Ｖ、5Ｖ）を供給できますので、その範囲のハードウエアであれば特に電源を用意する必要がありません。電源のＯＮ，ＯＦＦはＨＥＷから行います。電源スイッチをオフにするか電池ははずして、Ｅ８ａからの電源でデバックするのが電池の消耗を気にせず良いでしょう。

以下省略

WindowsXPⓇ、WindowsVistⓇ、Windows7Ⓡはマイクロソフト社の登録商標です。
フォースⓇライタは弊社の登録商標です。

１．本文章に記載された内容は弊社有限会社ビーリバーエレクトロニクスの調査結果です。
２．本文章に記載された情報の内容、使用結果に対して弊社はいかなる責任も負いません。
３．本文章に記載された情報に誤記等問題がありましたらご一報いただけますと幸いです。
４．本文章は許可なく転載、複製することを堅くお断りいたします。

お問い合わせ先：
〒３５０－１２１３　埼玉県日高市高萩１１４１－１
TEL　０４２（９８５）６９８２
FAX　０４２（９８５）６７２０
Homepage：http//beriver.co.jp
e-mail：info@beriver.co.jp